Navigating the security landscape of generative AI

First published April 2, 2025

Last updated April 8, 2025

Table of Contents

Introduction

Generative artificial intelligence, specifically large language models (LLMs), is reshaping how organizations handle data, automate processes, and drive innovation. However, as these capabilities expand, they also expand current security risks and introduce new ones. Security frameworks and teams need to account for the new challenges that generative AI brings, such as context window overflow, agent mismanagement, and indirect prompt injections. As generative AI becomes a core technology within organizations, we also need to ensure that it's held to the same standard and compliance requirements as other technologies. Organizations that learn to take an agile approach to security will be well positioned in the marketplace as adoption of AI grows. This white paper provides an approach for CISOs to navigate these risks, offering detailed mitigation strategies, including enhanced input validation, real-time monitoring, and modular system architecture. We focused on eight initial threat vectors and have suggested mitigation strategies for each.

We view a strong security foundation as an accelerant to adopting generative AI that enables organizations to safely and confidently add it to their mix of technologies. While many current technologies can also help tighten security, generative AI brings a few additional nuances that must be addressed and are novel to security. Many of the recommendations in this paper are easier said than done, but augmenting technologies, both from AWS and our partners, are evolving to help address those gaps and should be considered.

Finally, this paper is intended to complement, and potentially reinforce, newly emerging generative AI security strategies such as OWASP Top 10 for LLM, MITRE ATLAS, and so on. AWS continues to participate in global standards bodies such as the Coalition for Secure AI (CoSAI), Frontier Model Forum, and more to provide insights.

The following challenges represent a prescriptive point of view from the AWS proactive security team.

Regulatory and standards evolution

Global interest has increased among regulators given the potential ramifications of improper uses of generative AI. The EU AI Act is one of the better-known regulations, and it predominately takes a risk-based approach. High-risk applications, such as law enforcement, healthcare, and workloads impacting human rights are given a higher

regulatory bar to meet. This can include clauses such as including human-in-the-loop or even an outright prohibition of a workload.

A risk-based approach strikes an effective balance between industry conditions and regulatory needs. On one hand, there are risks to trusting the outputs of an LLM for a life-critical workload. However, a joke-telling chatbot should not be held to the same standards.

Legal precedent is expected to shape regulatory actions in concert with outputs from standards agencies such as NIST. In the long term, a patchwork quilt of regulations will likely emerge in the US while other countries that have previously aligned with the GDPR will likely align with the EU AI Act.

Certain compliance standards such as ISO42001 and IRAP have started to cover AI security. HITRUST is also building AI controls. There is the potential that the EU will accept ISO42001 as an effective risk management practice. However, EU regulatory frameworks continue to evolve, as demonstrated by the SHREMS II decision regarding GDPR.

Organizations are encouraged to use NIST ahead of regulatory actions and take an agile approach to their security posture. Organizations that stay ahead of compliance and regulatory frameworks by taking a security-first approach will have a competitive advantage within the marketplace once regulations begin to take hold.

Generative AI's impact on organization structures

The impact of generative AI depends on the current organizational structures. Traditionally, there have been tensions between data science teams and security teams. Data science often needs broad access to data while security strives for a least-privilege approach.

Organizations that follow a methodology of scaling security instead of consolidating it into a single organizational structure will be better positioned for success. A scaled approach creates a culture of security and helps security leaders focus on core issues. One example of a scaled approach is the AWS Security Guardians program. This program trains Amazon staff how to do security reviews, collaborate with teams on taking a security-first approach, and identify when to escalate to security engineering.

An organization can take a similar approach and embed security into its data science teams, called shifting security left. This keeps security close to the work, allowing for fast feedback. When taking this approach, it's important to take the approach of enablement instead of simply blocking work from happening. While it's easier to say "no," a better approach is to think of how to say "yes, but."

Technical organizations that invest in a scaled security approach also see an increase in software delivery velocity, because security reviews are traditionally gatekept by a central team. There is an organizational tax imposed whenever a team moves between organizational structures. This tax can be reduced by keeping a tight feedback loop with security.

Scoping generative AI use cases

The first step in developing a robust security strategy for generative AI is to properly scope its use within your organization. See the AWS Generative AI Security Scoping Matrix (shown in the following figure) to categorize your use cases.

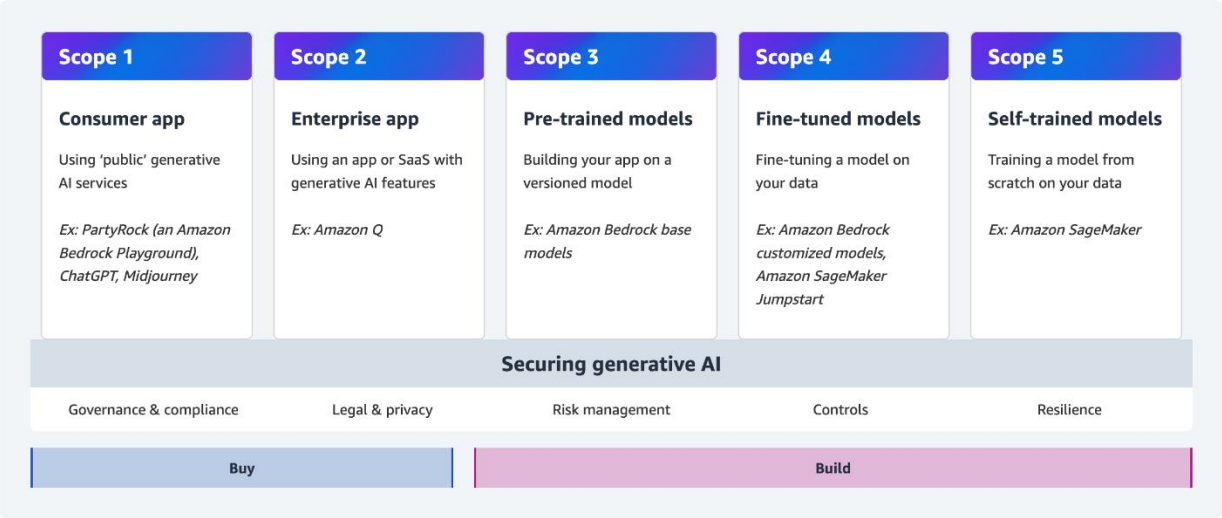

Figure 1: Generative AI Security Scoping Matrix, a mental model to classify use cases.

The scoping matrix includes five scopes. For Scope 1 or Scope 2 applications, which typically involve off-the-shelf AI solutions, adopt a buyer's perspective. Focus on risk management through data governance and carefully review enterprise agreements. It's crucial to clearly understand which data is authorized for sharing and under what circumstances. While Scope 2 applications would typically be built to support

enterprise data security and compliance needs, Scope 1 applications most often are not.

For Scope 3, 4, or 5 applications, which involve more customized or internally developed AI solutions, adopt a builder's perspective. Determine what data is in scope for the application and conduct thorough threat modeling (detailed in Threat modeling for generative AI applications). In these three scopes, while you generally have more control over your data, you also have more responsibility in protecting it. Be aware that the complexities for managing both the model and data components progressively increase as you move from Scope 3 through Scope 5, requiring increasingly rigorous security considerations at each level.

For all scopes, the way you approach governance and compliance, legal and privacy, risk management, controls, and resilience requirements will vary. However, by understanding the scopes that align to your use cases, you can quickly narrow down how you will address the requirements that align to these different security dimensions.

Optimizing generative AI security and responsible AI

Understanding the distinct yet complementary roles of generative AI security and responsible AI is essential for comprehensive risk management. While security safeguards systems and data assets, responsible AI addresses broader safety imperatives including bias prevention, output reliability, and ethical considerations. Both security and responsible AI controls must be integrated throughout the entire AI system lifecycle within an organization.

Traditional security controls focused on perimeter protection and data access are necessary but insufficient for generative AI systems, which face unique threat vectors such as prompt injection, model poisoning, and adversarial exploits. This new landscape requires innovative security approaches specifically designed for AI architectures.

Calibrate your risk strategy based on deployment context and user exposure. For example, internal enterprise applications warrant different controls compared to public-facing AI systems. Define specific thresholds for both security risks (such as data exposure) and AI safety risks (including bias, harmful content generation, and hallucinations). Given the probabilistic nature of generative AI outputs and associated

risks, safety controls might need more stringent thresholds than traditional security measures. Align your risk framework with established standards like the NIST AI Risk Management Framework (RMF) while adapting controls for AI-specific challenges.

Threat modeling for generative AI applications

Threats on AI and machine learning (AI/ML) systems are becoming more frequent, moving beyond controlled environments to real-world production deployments. These threats target vulnerabilities such as exposure to personally identifiable information (PII), lack of oversight in decision-making, and insufficient logging and monitoring.

Conducting a thorough threat model for your generative AI application is essential. Begin by defining the level of agency that you will provide to the LLM and any AI agents that you might use. This involves determining the extent of autonomy and decision-making power the AI system will have.

Next, clearly define where authentication and authorization should be performed. For guidance on this, see this blog post. Align your threat modeling process with established web security and generative AI frameworks such as MITRE ATLAS and OWASP Top 10 for LLMs. These frameworks provide comprehensive guidance on potential threats and mitigation strategies specific to AI systems.

Implement applicable traditional security controls for data security and deploy AI-specific mitigations for AI safety risks. For example, consider implementing controls such as Amazon Bedrock Guardrails to minimize the possibility of your AI-based application generating harmful or biased content. However, it's important to note that traditional controls like perimeter protection don't extend to cover many of the new threat vectors such as model-specific protections (see Traditional Cybersecurity Controls DO NOT STOP Attacks Against AI). Make sure that you layer traditional controls with emerging controls and capabilities that are designed to address the unique requirements for LLMs and the systems built around them.

Carefully consider the pros and cons of logging in generative AI systems and determine the appropriate level of logging for your application. This decision will directly impact your ability to monitor, audit, and respond to incidents.

Finally, establish incident response plans that align with your logging capabilities and the specific risks associated with your generative AI application.

Managing data access for data science teams

Data science teams need access to real-world data to do their jobs. Deidentification of production data is one approach that you can use to reduce the risk of improper data handling. However, deidentification can be challenging. The United States and other countries have differing definitions of PII. For example, in the EU, if a person can be re-identified, then the GDPR doesn't recognize the data as deidentified.

Synthetic data is another approach, but this might have lower fidelity. The debate is ongoing on whether we will run into a peak data scenario that impacts effective scaling of generative AI. However, new approaches to generating synthetic data are increasingly showing efficacy for generative AI applications. In some cases, using generative AI to create synthetic data risks biasing the output of synthetic data based on the bias of the data the generative AI model was trained on.

One approach is to use auditing. Auditing has been successfully used in healthcare to protect protected health information (PHI). Care teams often need access to sensitive data to do their jobs. Preventing access can have safety ramifications, but still needs to be balanced with security. A similar approach can be used to maintain appropriate use of data. Automation can be applied to audit logs that can identify anomalous behavior. This has been successful in large health systems where it has identified care team members who were accessing data they should not have.

Finally, keeping data contained in the cloud where the security team maintains the environment is a way to protect against data leaks. After data ends up on a local workstation, controlling access to it becomes almost impossible, even with data loss prevention tools. When appropriate controls are applied, the cloud can adhere to the most stringent security requirements.

Shadow generative AI

The productivity improvements of generative AI cannot be understated. When organizations outright ban generative AI or are slow to adopt it, employees will find ways to use consumer grade (scope 1) applications. This has led to an explosion of

shadow generative AI in organizations, creating unmanaged and uncontrolled data risks. A dual approach needs to be taken here:

- **Provide approved tooling to the workforce.** By offering sanctioned AI tools, organizations can reduce shadow AI usage while simultaneously improving visibility into how generative AI is being used.
- **Build out sufficient observability in the organization.** Organizations should invest in security lakes and AI monitoring dashboards to track violations of corporate AI policies. This includes monitoring active models, costs, prompt inputs and outputs, and the enforcement of security guardrails. Endpoint monitoring solutions should be deployed to detect unauthorized use of shadow generative AI, providing a better compliance and security posture.

Continual evaluation and best practices

Given the non-deterministic nature of generative AI technologies, implementing a strategy for continual evaluation is crucial. Regularly review the accuracy of your AI systems and maintain ongoing compliance with the established safety and security parameters outlined in your initial assessments.

Organizations should also follow best practices for prompt engineering such as those detailed in this workshop.

Threats and mitigations

In this section, we discuss the main threats we encountered and mitigation strategies.

Context window overflow

LLMs process a limited amount of information within a fixed context window. Exceeding this limit can cause the model to forget earlier instructions, which adversaries can exploit by flooding the model with excessive or malicious content. This can result in unpredictable system behavior, data leaks, and unauthorized actions.

Mitigation strategies:

- **Input management.** Limit the size of the input going into the model. Prioritize essential information and sanitize potentially harmful or excessive inputs before they reach the model.
- **Real-time monitoring.** Use monitoring systems that trigger alerts when the context window is nearing capacity. Proactively manage context overflow by truncating unessential data.

Agent vulnerabilities

Agents extend AI functionality, but are vulnerable to exploitation if not adequately secured. These vulnerabilities can result in unauthorized access, data breaches, and compromised external integrations, which expose sensitive information.

Mitigation strategies:

- **Principle of least privilege.** Implement least privilege for all agents and external integrations to reduce the potential exploit surface.
- **Regular audits and patching.** Enforce continuous audits, code reviews, and the application of security patches to help protect against known and emerging vulnerabilities.
- **Agent isolation.** Isolate agents to help prevent them from directly accessing sensitive parts of the system, using sandboxing techniques to minimize the impact of compromised agents.

Indirect prompt injections

Indirect prompt injections occur when adversaries embed malicious commands within seemingly benign user inputs. The AI system might inadvertently execute these instructions, resulting in unauthorized outputs or data manipulation.

Mitigation strategies:

- **Advanced input validation.** Use context-aware input filters to detect and neutralize malicious instructions embedded in user inputs. Traditional methods such as using a WAF do not go far enough. Specialized models trained on potential inputs might be needed to sufficiently mitigate this issue.

- **Layered defenses.** Implement multi-level checks where inputs are scrutinized at several stages to detect abnormalities.
- **User education.** Train administrators and users to identify the signs of prompt injections and respond promptly to security breaches.

Enhance safeguards for adversarial exploits

Traditional AI safeguards, such as basic content moderation, struggle against sophisticated exploits that use encoded instructions to bypass filters. This necessitates a rethinking of how AI systems are safeguarded against adversarial exploitation.

Mitigation strategies:

- **Contextual filters.** Go beyond basic keyword detection by using filters that assess the context of inputs to catch more nuanced adversarial techniques.
- **Adaptive defenses.** Incorporate machine learning-powered filters that continuously learn and adapt to new adversarial techniques.
- **Defense-in-depth.** Introduce layered security mechanisms, such as refusal classifiers and real-time input monitoring, to fortify AI system integrity.

Trust and security boundaries

Establishing and managing trust boundaries in AI applications is essential to help prevent unauthorized access and safeguard sensitive data. Rigorous data flow analysis can identify weak points where sensitive information could be inadvertently exposed.

Best practices:

- **Data classification.** Make sure that your data has been properly classified according to sensitivity.
- **Data flow mapping.** Conduct comprehensive analyses of data flows from input to output to make sure that sensitive data is appropriately safeguarded.
- **Principle of least privilege.** Make sure that users, agents, and external integrations have the minimal access rights required for their tasks.
- **Secure APIs.** Secure API endpoints through robust authentication, authorization, and continuous input validation.

- **Data hygiene.** Introduce standard operating procedures for cleaning and validating data.

Design AI systems for reliability

LLMs, despite their efficiency, can introduce reliability risks that should be addressed up front. Designing systems to minimize risks such as model failures or adversary-controlled outputs should be a key design consideration.

Strategies for resilience:

- **Modular architecture.** Adopt a modular system architecture that decouples critical components, allowing isolation of faults and failures.
- **Validation layers.** Use multiple validation layers to assess model outputs for plausibility and consistency before they reach end users.
- **Human oversight.** Implement human-in-the-loop systems for reviewing critical decisions and low-confidence outputs to reduce potential errors.

Isolate sensitive data from AI models

Generative AI systems, particularly LLMs, are at risk for data extraction and leakage, especially when handling sensitive information. Implementing strict data isolation strategies is crucial to help prevent confidential information from being exposed through prompts or model outputs.

Data isolation strategies:

- **Data minimization.** Limit the data exposed to the model, providing only what is necessary for the task at hand.
- **Differential privacy.** Employ differential privacy techniques to make sure that sensitive data cannot be reconstructed from model outputs.
- **Secure prompt engineering.** Do not include sensitive data in prompts, and verify secure data handling by third-party services.
- **Use Retrieval Augmented Generation (RAG).** Use RAG with strong AuthZ and AuthN to augment model data over fine tuning.

Minimize data leaks from overprivileged agents, logging, and caching

Overprivileged agents and improper handling of logs or cached data can lead to serious security breaches, allowing adversaries to access sensitive information or manipulate outputs.

Preventive measures:

- **Access control.** Limit agent access using strict role-based access control (RBAC).
- **Anonymized logging.** Make sure that logs do not inadvertently capture sensitive information. Use anonymization techniques when necessary.
- **Secure caching.** Encrypt cached data and enforce strict expiration policies to help prevent unauthorized access to sensitive cached information.

Conclusion

The deployment of LLMs and generative AI systems requires an agile security approach to make sure that adopting generative AI is a business accelerant. For CISOs, addressing these risks requires a multi-layered approach to security, emphasizing robust input validation, continuous monitoring, modular system architecture, and enhanced data safeguarding techniques. By implementing these strategies, organizations can capitalize on the transformative potential of AI technologies while helping customers keep sensitive data secure, mitigate adversarial risks, and address regulatory compliance requirements.

About the authors

Matthew Schwartz is a Principal Security Engineer at Amazon specializing in generative AI security and risk management. With over 20 years of experience, he helps organizations implement strategic security frameworks that enable AI integration while maintaining compliance standards, leveraging his deep expertise in cloud computing and digital transformation to protect critical assets in an increasingly AI-driven landscape.

Mac Stevens is a Senior Solutions Architect with the AWS Public Sector Team. With a background as a security leader, he brings extensive experience in addressing emerging risks. Mac specializes in generative AI security, focusing on helping customers incorporate AI technologies while maintaining robust security measures.

Mac is also one of the investigators for the NIST AI Safety & Security Institute. As a passionate builder, he loves identifying innovative ways to help customers apply technology to both business and security challenges.

Thank you to the following for their contributions: Paul Vixie, Jessica Kropf, Hart Rossman, Phillip Simpson, Mark Ryland, and Matt Saner.

Written with support from partners: Accenture, Arctic Wolf, Checkmarx, Check Point, Crowdstrike, Datadog, F5, Fortinet, Hidden Layer, Netskope, Orca, PwC, Query.ai, and Snyk.